Contents & Goals

Last Lecture:

• Started location reachability decidability (by region construction)

This Lecture:

• **Educational Objectives:** Capabilities for following tasks/questions.
  • What is a region? What is the region automaton of this TA?
  • What’s the time abstract system of a TA? Why did we consider this?
  • What can you say about the complexity of Region-automaton based reachability analysis?
  • What’s a zone? In contrast to a region?
  • Motivation for having zones?
  • What’s a DBM? Who needs to know DBMs?

• **Content:**
  • Region automaton cont’d
  • Reachability Problems for Extended Timed Automata
  • Zones
  • Difference Bound Matrices
The Location Reachability Problem Cont’d
Definition 4.29. [Region Automaton] The region automaton \( \mathcal{R}(\mathcal{A}) \) of the timed automaton \( \mathcal{A} \) is the labelled transition system

\[
\mathcal{R}(\mathcal{A}) = (\text{Conf}(\mathcal{R}(\mathcal{A})), B^?, \{ \xrightarrow{\alpha}_{\mathcal{R}(\mathcal{A})} | \alpha \in B^? \}, C_{\text{ini}})
\]

where

- \( \text{Conf}(\mathcal{R}(\mathcal{A})) = \{ \langle \ell, [\nu] \rangle | \ell \in L, \nu : X \rightarrow \text{Time}, \nu \models I(\ell) \} \),
- for each \( \alpha \in B^? \),
  \[
  \langle \ell, [\nu] \rangle \xrightarrow{\alpha}_{\mathcal{R}(\mathcal{A})} \langle \ell', [\nu'] \rangle \text{ if and only if } \langle \ell, \nu \rangle \xrightarrow{\alpha} \langle \ell', \nu' \rangle \text{ in } \mathcal{U}(\mathcal{A}),
  \]
- \( C_{\text{ini}} = \{ \langle \ell_{\text{ini}}, [\nu_{\text{ini}}] \rangle \} \cap \text{Conf}(\mathcal{R}(\mathcal{A})) \) with \( \nu_{\text{ini}}(X) = \{0\} \).

Proposition. The transition relation of \( \mathcal{R}(\mathcal{A}) \) is well-defined, that is, independent of the choice of the representative \( \nu \) of a region \( [\nu] \).
Example: Region Automaton

\( U(A): \)

\[ \langle \text{light, } [x = 0] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{bright, } [x = 0] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{bright, } [x = 0.1] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{bright, } [x = 1.0] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{bright, } [x = 3.0] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{bright, } [x = 3.001] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{off, } [x = 0] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{off, } [x = 2.9] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{off, } [x = 3.0] \rangle \xrightarrow{\text{press}} \cdots \]

\[ \langle \text{off, } [x = 3.001] \rangle \xrightarrow{\text{press}} \cdots \]
Remark 4.30. That a configuration $\langle \ell, [\nu] \rangle$ is reachable in $\mathcal{R}(A)$ represents the fact, that all $\langle \ell, \nu \rangle$ are reachable.

IAW: in $A$, we can observe $\nu$ when
location $\ell$ has just been entered.

The clock values reachable by staying/letting time pass in $\ell$ are not explicitly represented by the regions of $\mathcal{R}(A)$. 
Decidability of The Location Reachability Problem

**Claim:** (Theorem 4.33)

The location reachability problem is **decidable** for timed automata.

**Approach:** Constructive proof.

- ✓ Observe: clock constraints are **simple**
  — w.l.o.g. assume constants $c \in \mathbb{N}_0$.

- ✓ **Def. 4.19**: time-abstract transition system $\mathcal{U}(\mathcal{A})$ — abstracts from uncountably many delay transitions, still infinite-state.

- ✓ **Lem. 4.20**: location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.

- ✓ **Def. 4.29**: region automaton $\mathcal{R}(\mathcal{A})$ — equivalent configurations collapse into regions

- ✗ **Lem. 4.32**: location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.

- ✗ **Lem. 4.28**: $\mathcal{R}(\mathcal{A})$ is **finite**.
Lemma 4.32. [Correctness] For all locations $\ell$ of a given timed automaton $A$ the following holds:

$\ell$ is reachable in $U(A)$ if and only if $\ell$ is reachable in $R(A)$.

For the Proof:

Definition 4.21. [Bisimulation] An equivalence relation $\sim$ on valuations is a (strong) bisimulation if and only if, whenever

$v_1 \sim v_2$ and $\langle \ell, v_1 \rangle \xrightarrow{\alpha} \langle \ell', v'_1 \rangle$

then there exists $v'_2$ with $v'_1 \sim v'_2$ and $\langle \ell, v_2 \rangle \xrightarrow{\alpha} \langle \ell', v'_2 \rangle$.

Lemma 4.26. [Bisimulation] $\Rightarrow$ is a strong bisimulation.
Claim: (Theorem 4.33)

The location reachability problem is **decidable** for timed automata.

**Approach:** Constructive proof.

✔ Observe: clock constraints are **simple**
— w.l.o.g. assume constants $c \in \mathbb{N}_0$.

✔ **Def. 4.19:** time-abstract transition system $\mathcal{U}(\mathcal{A})$ — abstracts from uncountably many delay transitions, still infinite-state.

✔ **Lem. 4.20:** location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.

✔ **Def. 4.29:** region automaton $\mathcal{R}(\mathcal{A})$ — equivalent configurations collapse into regions

✔ **Lem. 4.32:** location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.

✘ **Lem. 4.28:** $\mathcal{R}(\mathcal{A})$ is **finite**.
Lemma 4.28. Let $X$ be a set of clocks, $c_x \in \mathbb{N}_0$ the maximal constant for each $x \in X$, and $c = \max\{c_x \mid x \in X\}$. Then
\[
(2c + 2)^{|X|} \cdot (4c + 3)^{\frac{1}{2}}|X| \cdot (|X| - 1)
\]
is an upper bound on the number of regions.

Proof: [Olderog and Dierks, 2008]
Observations Regarding the Number of Regions

- Lemma 4.28 **in particular** tells us that each timed automaton (in our definition) has **finitely** many regions.

\[ \text{Thus } R(A) \text{ is finite} \]

- Note: the upper bound is a **worst case**, not an **exact bound**.

  e.g. if \( c_x < c_y \), 4.28 still works with \( c = \max \{ c_x, c_y \} \)
Decidability of The Location Reachability Problem

Claim: (Theorem 4.33)

The location reachability problem is decidable for timed automata.

Approach: Constructive proof.

✔ Observe: clock constraints are simple
— w.l.o.g. assume constants \( c \in \mathbb{N}_0 \).

✔ Def. 4.19: time-abstract transition system \( \mathcal{U}(\mathcal{A}) \) — abstracts from uncountably many delay transitions, still infinite-state.

✔ Lem. 4.20: location reachability of \( \mathcal{A} \) is preserved in \( \mathcal{U}(\mathcal{A}) \).

✔ Def. 4.29: region automaton \( \mathcal{R}(\mathcal{A}) \) — equivalent configurations collapse into regions

✔ Lem. 4.32: location reachability of \( \mathcal{U}(\mathcal{A}) \) is preserved in \( \mathcal{R}(\mathcal{A}) \).

✔ Lem. 4.28: \( \mathcal{R}(\mathcal{A}) \) is finite.
Putting It All Together

Let $A = (L, B, X, I, E, \ell_{\text{init}})$ be a timed automaton, $\ell \in L$ a location.

- $\mathcal{R}(A)$ can be constructed effectively.
- There are finitely many locations in $L$ (by definition).
- There are finitely many regions by Lemma 4.28.
- So $\text{Conf}(\mathcal{R}(A))$ is finite (by construction).
- It is decidable whether ($C_{\text{init}}$ of $\mathcal{R}(A)$ is empty) or whether there exists a sequence

$$\langle \ell_{\text{init}}, [\nu_{\text{init}}] \rangle \xrightarrow{\alpha} \mathcal{R}(A) \langle \ell_1, [\nu_1] \rangle \xrightarrow{\alpha} \mathcal{R}(A) \cdots \xrightarrow{\alpha} \mathcal{R}(A) \langle \ell_n, [\nu_n] \rangle$$

such that $\ell_n = \ell$ (reachability in graphs).

So we have

**Theorem 4.33. [Decidability]**
The location reachability problem for timed automata is **decidable**.
The Constraint Reachability Problem

- **Given:** A timed automaton $A$, one of its control locations $\ell$, and a clock constraint $\phi$.

- **Question:** Is a configuration $\langle \ell, \nu \rangle$ reachable where $\nu \models \phi$, i.e. is there a transition sequence of the form

$$
\langle \ell_{ini}, \nu_{ini} \rangle \xrightarrow{\lambda_1} \langle \ell_1, \nu_1 \rangle \xrightarrow{\lambda_2} \langle \ell_2, \nu_2 \rangle \xrightarrow{\lambda_3} \ldots \xrightarrow{\lambda_n} \langle \ell_n, \nu_n \rangle = \langle \ell, \nu \rangle
$$

in the labelled transition system $\mathcal{T}(A)$ with $\nu \models \phi$?

- **Note:** we just observed that $R(A)$ loses some information about the clock valuations that are possible in/from a region.

**Theorem 4.34.** The constraint reachability problem for timed automata is decidable.
The Delay Operation

- Let $[\nu]$ be a clock region.
- We set

$$delay[\nu] = \{\nu' + t \mid \nu' \cong \nu \text{ and } t \in \text{Time}\}.$$
The Delay Operation

- Let $[\nu]$ be a clock region.
- We set

$$delay[\nu] = \{\nu' + t \mid \nu' \equiv \nu \text{ and } t \in \text{Time}\}.$$  

- Note: $delay[\nu]$ can be represented as a finite union of regions. 

For example, with our two-clock example we have

$$delay[x = y = 0] = [x = y = 0] \cup [0 < x = y < 1] \cup [x = y = 1] \cup [1 < x = y].$$
Zones

(Presentation following [Fränzle, 2007])
Lemma 4.28. Let $X$ be a set of clocks, $c_x \in \mathbb{N}_0$ the maximal constant for each $x \in X$, and $c = \max\{c_x \mid x \in X\}$. Then
\[(2c + 2)^{|X|} \cdot (4c + 3)^{\frac{1}{2}|X| \cdot (|X| - 1)}\]
is an upper bound on the number of regions.

- In the desk lamp controller,

![Diagram of a desk lamp controller with states off, light, and bright. The diagram shows transitions based on pressing a button, with conditions such as $x := 0$, $x \leq 3$, and $x > 3$.]

many
all regions are reachable in $\mathcal{R}(\mathcal{L})$, but we convinced ourselves that it’s actually only important whether $\nu(x) \in [0, 3]$ or $\nu(x) \in (3, \infty)$. So: seems there are even equivalence classes of undistinguishable regions.
Wanted: Zones instead of Regions

- In $\mathcal{R}(\mathcal{L})$ we have transitions:
  - $\langle \text{light}, \{0\} \rangle \xrightarrow{\text{press?}} \langle \text{bright}, \{0\} \rangle$, $\langle \text{light}, \{0\} \rangle \xrightarrow{\text{press?}} \langle \text{bright}, (0, 1) \rangle$,
  - $\langle \text{light}, \{0\} \rangle \xrightarrow{\text{press?}} \langle \text{bright}, (2, 3) \rangle$, $\langle \text{light}, \{0\} \rangle \xrightarrow{\text{press?}} \langle \text{bright}, \{3\} \rangle$.

- Which seems to be a complicated way to write just:
  $$\langle \text{light}, \{0\} \rangle \xrightarrow{\text{press?}} \langle \text{bright}, [0, 3] \rangle$$

- Can't we constructively abstract $\mathcal{L}$ to:

$$\begin{align*}
\langle \text{off}, \{0\} \rangle & \xrightarrow{\text{press?}} \langle \text{light}, \{0\} \rangle \xrightarrow{\text{press?}} \langle \text{bright}, [0, 3] \rangle \\
\langle \text{off}, (3, \infty) \rangle & \xrightarrow{\text{press?}} \langle \text{off}, [0, \infty) \rangle
\end{align*}$$
What is a Zone?

**Definition.** A *(clock) zone* is a set $z \subseteq (X \rightarrow \text{Time})$ of valuations of clocks $X$ such that there exists $\varphi \in \Phi(X)$ with

$$\nu \in z \text{ if and only if } \nu \models \varphi.$$ 

**Example:**

is a clock zone by

$$\varphi = (x \leq 2) \land (x > 1) \land (y \geq 1) \land (y < 2) \land (x - y \geq 0)^0$$
What is a Zone?

Definition. A (clock) zone is a set $z \subseteq (X \rightarrow \text{Time})$ of valuations of clocks $X$ such that there exists $\varphi \in \Phi(X)$ with

$$\nu \in z \text{ if and only if } \nu \models \varphi.$$ 

Example:

is a clock zone by

$$\varphi = (x \leq 2) \land (x > 1) \land (y \geq 1) \land (y < 2) \land (x - y \geq 0)$$

- Note: Each clock constraint $\varphi$ is a **symbolic representation** of a zone.
- But: There's no one-on-one correspondence between clock constraints and zones. The zone $z = \emptyset$ corresponds to $(x > 1 \land x < 1)$, $(x > 2 \land x < 2)$, ...
More Examples: Zone or Not?

YES by
\[(x \geq 1) \land (y \geq 1) \land (x - y \geq 0) \land (x - y \leq 2)\]

YES by
\[(x > 1) \land (x \leq 2) \land (y = 0)\]

\((x = y = 1) \lor (x = 1 \land y = 2) \land \text{NO}\)
Zone-based Reachability

Given:

- • off
- • light \( x := 0 \)
- • press \( x \leq 3 \)
- • bright \( x > 3 \)

and initial configuration \( \langle \text{off}, \{0\} \rangle \)

Assume a function

\[ \text{Post}_e : (L \times \text{Zones}) \rightarrow (L \times \text{Zones}) \]

such that \( \text{Post}_e(\langle \ell, z \rangle) \) yields the configuration \( \langle \ell', z' \rangle \) such that

- zone \( z' \) denotes exactly those clock valuations \( \nu' \)
- which are reachable from a configuration \( \langle \ell, \nu \rangle, \nu \in z \),
- by taking edge \( e = (\ell, \alpha, \varphi, Y, \ell') \in E \).

\[ \text{Press} \text{ } \text{edge of the automaton} \]
Zone-based Reachability

Given:

- \( \text{off} \) \( \xrightarrow{\text{press?}} \) \( \text{light} \): \( x := 0 \)
- \( \text{light} \) \( \xrightarrow{\text{press?}} \) \( \text{bright} \): \( x \leq 3 \)
- \( \text{light} \) \( \xrightarrow{\text{press?}} \) \( \text{off} \): \( x > 3 \)

and initial configuration \( \langle \text{off}, \{0\} \rangle \)

Assume a function

\[
\text{Post}_e : (L \times \text{Zones}) \rightarrow (L \times \text{Zones})
\]

such that \( \text{Post}_e(\langle \ell, z \rangle) \) yields the configuration \( \langle \ell', z' \rangle \) such that

- zone \( z' \) denotes exactly those clock valuations \( \nu' \)
- which are reachable from a configuration \( \langle \ell, \nu \rangle, \nu \in z \),
- by taking edge \( e = (\ell, \alpha, \varphi, Y, \ell') \in E \).

Then \( \ell \in L \) is reachable in \( A \) if and only if

\[
\text{Post}_{e_n} (\ldots (\text{Post}_{e_1}(\langle \ell_{ini}, z_{ini} \rangle) \ldots ))
\]

for some \( e_1, \ldots, e_n \in E \).
Zone-based Reachability: In Other Words

Given:

- Press \( x := 0 \) if \( x \leq 3 \) and \( x > 3 \)

and initial configuration \( \langle \text{off}, \{0\} \rangle \)

Wanted: A procedure to compute the set

- \( \langle \text{light}, \{0\} \rangle \)
- \( \langle \text{bright}, [0, 3] \rangle \)
- \( \langle \text{off}, [0, \infty) \rangle \)

Set \( R := \{ \langle \text{ini}, z_{\text{ini}} \rangle \} \subset L \times \text{Zones} \)

Repeat

- pick a pair \( \langle \ell, z \rangle \) from \( R \) and an edge \( e \in E \) with source \( \ell \) such that \( \text{Post}_e(\langle \ell, z \rangle) \) is not already subsumed by \( R \)

- add \( \text{Post}_e(\langle \ell, z \rangle) \) to \( R \) until no more such \( \langle \ell, z \rangle \in R \) and \( e \in E \) are found.
Stocktaking: What’s Missing?

- Set $R := \{\langle \ell_{ini}, z_{ini} \rangle \} \subset L \times \text{Zones}$
- Repeat
  - pick
  - a pair $\langle \ell, z \rangle$ from $R$ and
  - an edge $e \in E$ with source $\ell$
    such that $\text{Post}_e(\langle \ell, z \rangle)$ is not already subsumed by $R$
  - add $\text{Post}_e(\langle \ell, z \rangle)$ to $R$
  until no more such $\langle \ell, z \rangle \in R$ and $e \in E$ are found.

Missing:

- Algorithm to effectively compute $\text{Post}_e(\langle \ell, z \rangle)$ for given configuration $\langle \ell, z \rangle \in L \times \text{Zones}$ and edge $e \in E$.
- Decision procedure for whether configuration $\langle \ell', z' \rangle$ is subsumed by a given subset of $L \times \text{Zones}$.

Note: Algorithm in general terminates only if we apply widening to zones, that is, roughly, to take maximal constants $c_x$ into account (not in lecture).
What is a Good “Post”? 

- If $z$ is given by a constraint $\varphi \in \Phi(X)$, then the zone component $z'$ of $\text{Post}_e(\ell, z) = \langle \ell', z' \rangle$ should also be a constraint from $\Phi(X)$. (Because sets of clock valuations are soo unhandily...) 

**Good news**: the following operations can be carried out by manipulating $\varphi$. 

- The *elapse time* operation: 

  $$\uparrow : \Phi(X) \rightarrow \Phi(X)$$

  Given a constraint $\varphi$, the constraint $\uparrow(\varphi)$, or $\varphi \uparrow$ in postfix notation, is supposed to denote the set of clock valuations 

  $$\{ \nu + t \mid \nu \models \varphi, t \in \text{Time} \}.$$ 

  In other symbols: we **want** 

  $$[[ \uparrow(\varphi) ]] = [[ \varphi \uparrow ]] = \{ \nu + t \mid \nu \in [[\varphi]], t \in \text{Time} \}.$$ 

  To this end: remove all upper bounds $x \leq c$, $x < c$ from $\varphi$ and add diagonals.
Good News Cont’d

Good news: the following operations can be carried out by manipulating $\varphi$.

- **elapse time** $\varphi \uparrow$ with
  \[
  \llbracket \varphi \uparrow \rrbracket = \{\nu + t \mid \nu \models \varphi, t \in \text{Time}\}
  \]

- **zone intersection** $\varphi_1 \land \varphi_2$ with
  \[
  \llbracket \varphi_1 \land \varphi_2 \rrbracket = \{\nu \mid \nu \models \varphi_1 \text{ and } \nu \models \varphi_2\}
  \]

- **clock hiding** $\exists x.\varphi$ with
  \[
  \llbracket \exists x.\varphi \rrbracket = \{\nu \mid \text{there is } t \in \text{Time such that } \nu[x := t] \models \varphi\}
  \]

- **clock reset** $\varphi[x := 0]$ with
  \[
  \llbracket \varphi[x := 0] \rrbracket = \llbracket x = 0 \land \exists x.\varphi \rrbracket
  \]
This is Good News...

...because given $\langle \ell, z \rangle = \langle \ell, \varphi_0 \rangle$ and $e = (\ell, \alpha, \varphi, \{y_1, \ldots, y_n\}, \ell') \in E$ we have

$$\text{Post}_e(\langle \ell, z \rangle) = \langle \ell', \varphi_5 \rangle$$

where

- $\varphi_1 = \varphi_0 \uparrow$
  
  let time elapse starting from $\varphi_0$: $\varphi_1$ represents all valuations reachable by waiting in $\ell$ for an arbitrary amount of time.

- $\varphi_2 = \varphi_1 \land I(\ell)$
  
  intersect with invariant of $\ell$: $\varphi_2$ represents the reachable good valuations.

- $\varphi_3 = \varphi_2 \land \varphi$
  
  intersect with guard: $\varphi_3$ are the reachable good valuations where $e$ is enabled.

- $\varphi_4 = \varphi_3[y_1 := 0] \ldots [y_n := 0]$
  
  reset clocks: $\varphi_4$ are all possible outcomes of taking $e$ from $\varphi_3$

- $\varphi_5 = \varphi_4 \land I(\ell')$
  
  intersect with invariant of $\ell'$: $\varphi_5$ are the good outcomes of taking $e$ from $\varphi_3$
Example

- $\varphi_1 = \varphi_0 \uparrow$  
  - let time elapse.
- $\varphi_2 = \varphi_1 \land I(\ell)$  
  - intersect with invariant of $\ell$
- $\varphi_3 = \varphi_2 \land \varphi$  
  - intersect with guard
- $\varphi_4 = \varphi_3[y_1 := 0] \ldots [y_n := 0]$  
  - reset clocks
- $\varphi_5 = \varphi_4 \land I(\ell')$  
  - intersect with invariant of $\ell'$

$$
\begin{array}{c}
\varphi_0 \\
\varphi_1 \\
\varphi_2 \\
\varphi_3 \\
\varphi_4 \\
\varphi_5
\end{array}
$$
Given a finite set of clocks $X$, a DBM over $X$ is a mapping

$$M : (X \cup \{x_0\} \times X \cup \{x_0\}) \rightarrow (\{<, \leq\} \times \mathbb{Z} \cup \{(<, \infty)\})$$

$M(x, y) = (\sim, c)$ encodes the conjunct $x - y \sim c$ ($x$ and $y$ can be $x_0$).
Difference Bound Matrices

- Given a finite set of clocks $X$, a DBM over $X$ is a mapping
  \[ M : (X \cup \{x_0\} \times X \cup \{x_0\}) \rightarrow (\{<, \leq\} \times \mathbb{Z} \cup \{(<, \infty)\}) \]
- $M(x, y) = (\sim, c)$ encodes the conjunct $x - y \sim c$ ($x$ and $y$ can be $x_0$).

- If $M$ and $N$ are DBM encoding $\varphi_1$ and $\varphi_2$ (representing zones $z_1$ and $z_2$), then we can efficiently compute $M \uparrow$, $M \land N$, $M[x := 0]$ such that
  - all three are again DBM,
  - $M \uparrow$ encodes $\varphi_1 \uparrow$,
  - $M \land N$ encodes $\varphi_1 \land \varphi_2$, and
  - $M[x := 0]$ encodes $\varphi_1[x := 0]$.

- And there is a **canonical form** of DBM — canonisation of DBM can be done in cubic time (**Floyd-Warshall** algorithm).

- Thus: we can define our ‘Post’ on DBM, and let our algorithm run on DBM.
Pros and cons

- **Zone-based** reachability analysis usually is explicit wrt. discrete locations:
  - maintains a list of location/zone pairs or
  - maintains a list of location/DBM pairs
  - **confined wrt. size of discrete state space**
  - avoids blowup by number of clocks and size of clock constraints through symbolic representation of clocks

- **Region-based** analysis provides a finite-state abstraction, amenable to finite-state symbolic MC
  - **less dependent on size of discrete state space**
  - exponential in number of clocks
References
References
